

A large, powerful blue wave is crashing over a surfer. The surfer is a person in a dark wetsuit, crouching on a light-colored surfboard, riding the base of the wave. The water is a deep blue, and the white foam of the wave is prominent. The overall scene is dynamic and energetic.

Own your ClickHouse data with Altinity.Cloud[®] Anywhere

Robert Hodges and Alexander Zaitsev

Let's make some introductions

Us

Database geeks with decades of experience in DBMS and applications

You

App developers looking to build real-time analytics to solve business problems*

* You also like Kubernetes ;)



ClickHouse support and services including [Altinity.Cloud](#)®
Authors of [Altinity Kubernetes Operator for ClickHouse](#)
and other open source projects

What's Altinity.Cloud Anywhere?

ClickHouse is a real-time analytic database

Understands SQL

Runs on bare metal to cloud

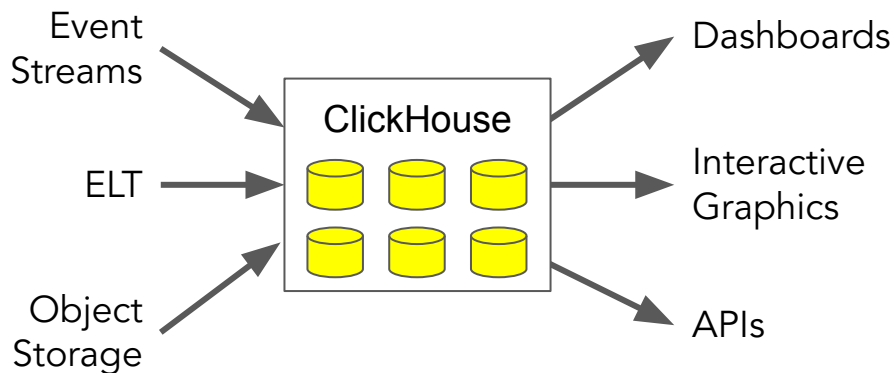
Shared nothing architecture

Stores data in columns

Parallel and vectorized execution

Scales to many petabytes

Is Open source (Apache 2.0)



It's the core engine for
low-latency analytics

Altinity.Cloud is a zero-maintenance SaaS for ClickHouse

Complete automation
of operations with
baked in DBA support



Cost and performance
optimized for real-time
analytics

Supports all versions and
features of ClickHouse
(also experimental ones!)

Run in any AWS/GCP
region or your own
Kubernetes clusters

Pure SaaS is convenient, but many users need more choices to meet business requirements

Run in your own VPCs for security compliance

Keep analytics close to on-prem data sources

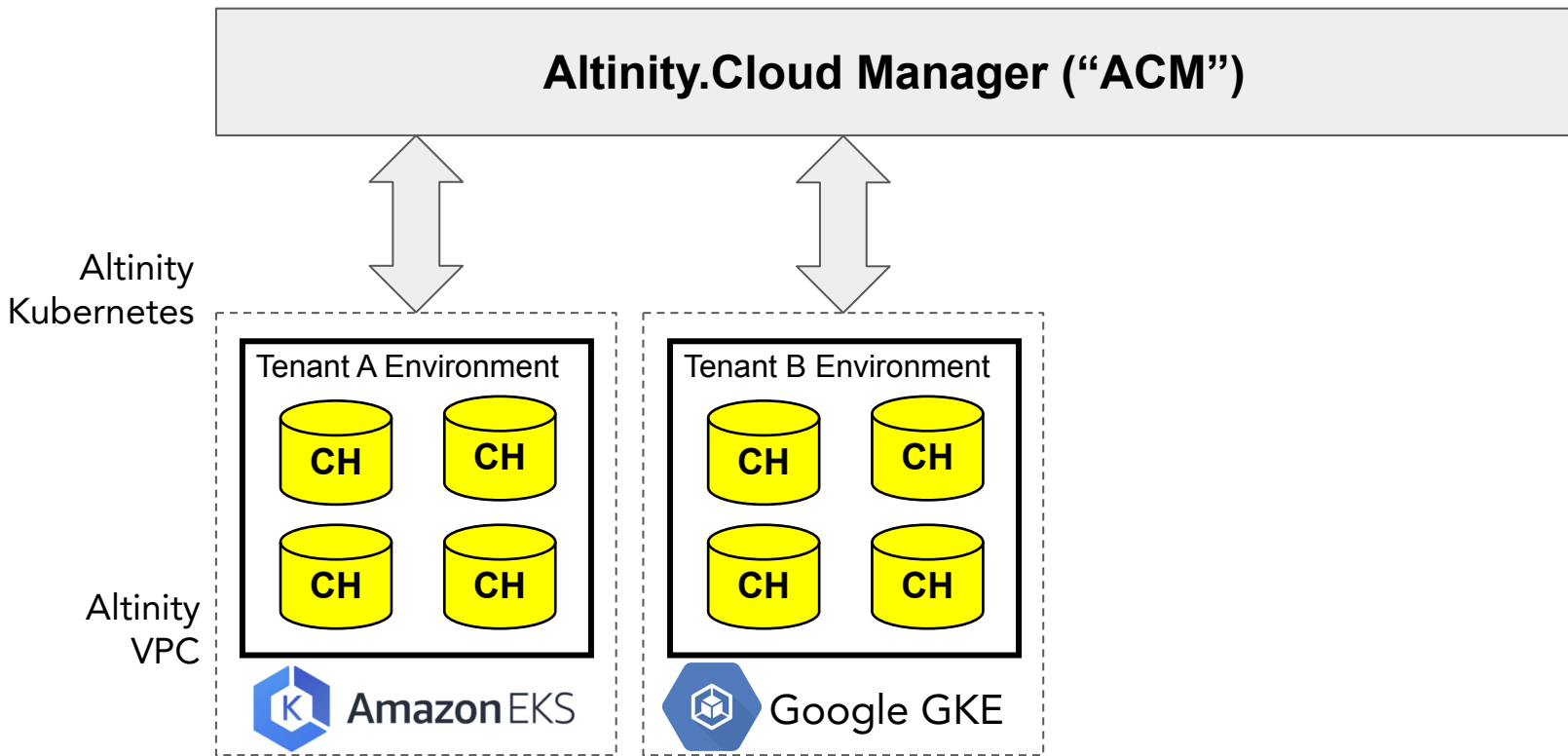


Use your own infrastructure to control costs

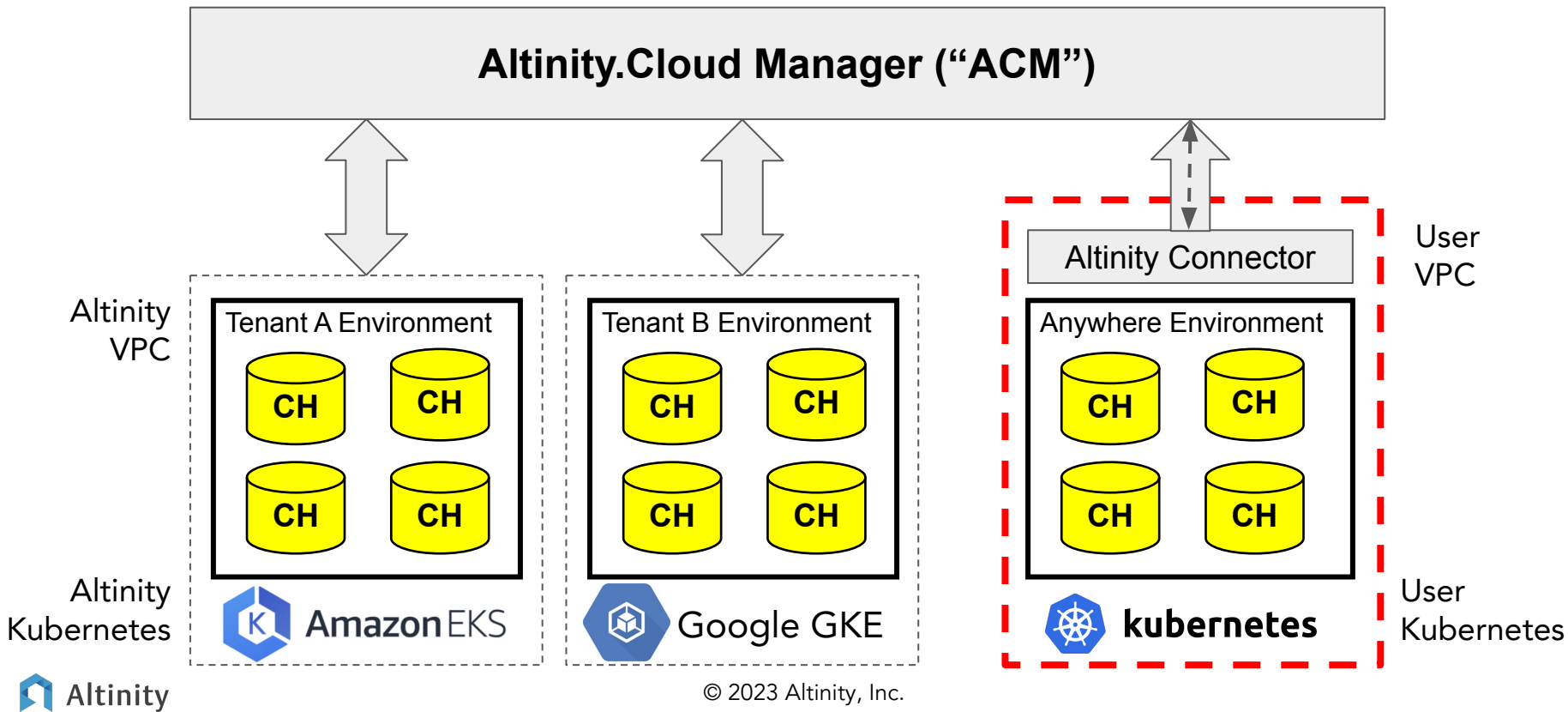
Use 100% open source to avoid vendor lock-in

DEMO TIME!

Altinity.Cloud runs ClickHouse in Kubernetes “environments”



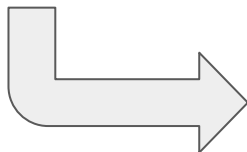
Altinity.Cloud Anywhere can run in your Kubernetes



Getting up and running with Altinity.Cloud Anywhere

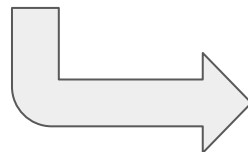
1

Prepare
Kubernetes



2

Connect to
Altinity.Cloud



3

Start your
clusters!

Preparing your Kubernetes

Choosing a Kubernetes distribution

Currently certified distributions

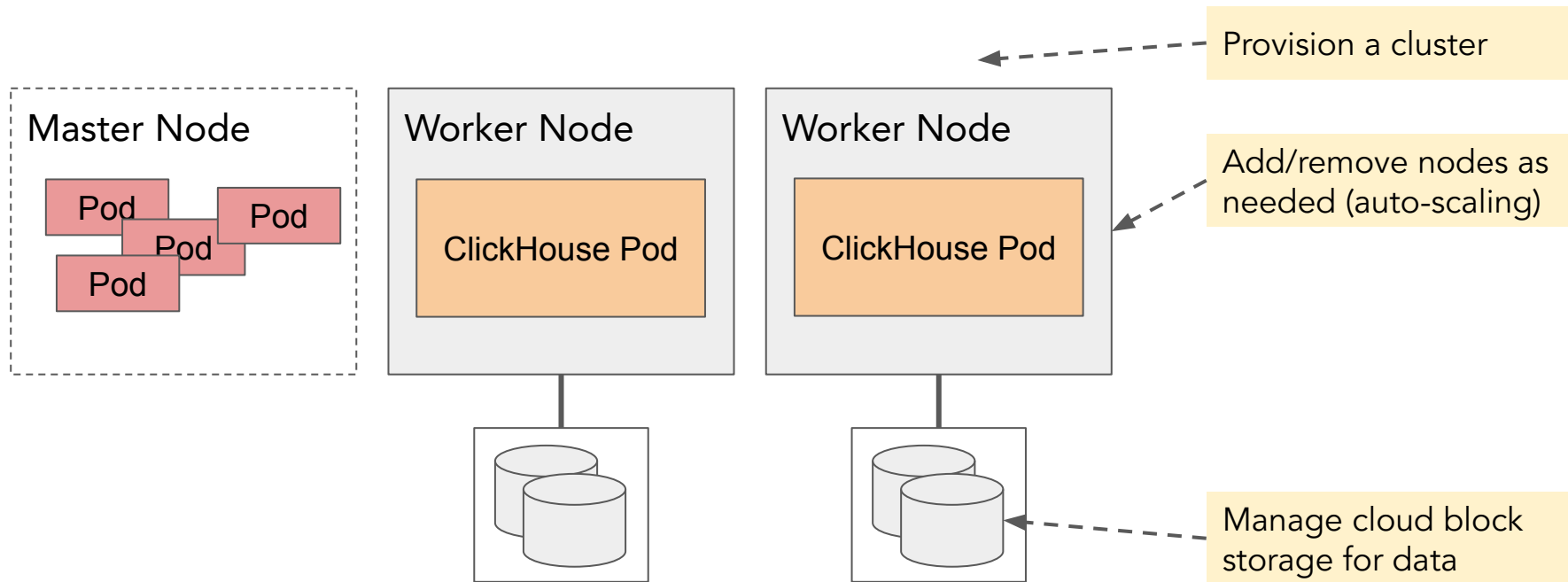


* Used for demo/test only

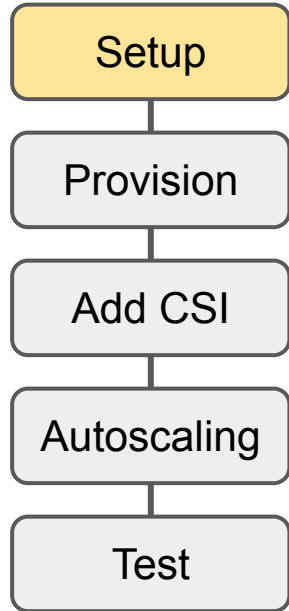
Pending certifications



Your Kubernetes setup needs to get a few things right

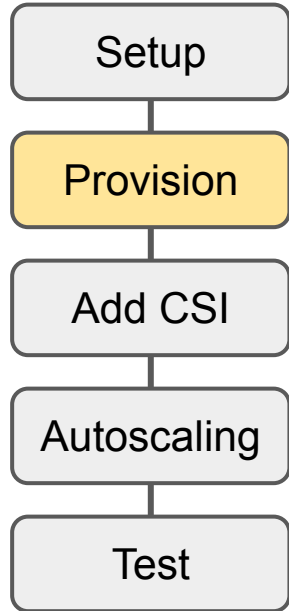


Setup to provision EKS Cluster using eksctl



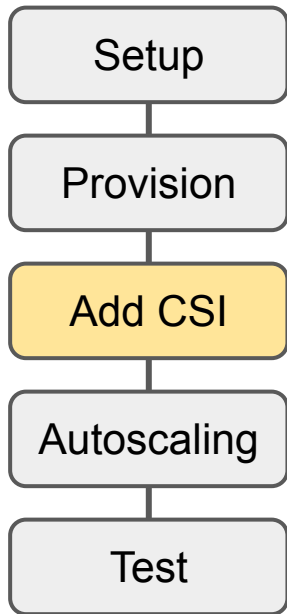
1. Bring up jump VM in AWS
2. Install [AWS CLI](#)
3. Install [eksctl](#)
4. Install [kubectl](#) (1.24 or greater)
5. Install [aws-iam-authenticator](#)
6. Add [AWS IAM permissions](#) to VM to create EKS clusters

Provision Kubernetes cluster on EKS



```
eksctl create cluster -f - << EOF
---
apiVersion: eksctl.io/v1alpha5
kind: ClusterConfig
metadata:
  name: ${CLUSTER_NAME}
  region: ${AWS_DEFAULT_REGION}
  version: "1.23"
  tags:
    karpenter.sh/discovery: ${CLUSTER_NAME}
managedNodeGroups:
  - instanceType: m5.large
    amiFamily: AmazonLinux2
    name: ${CLUSTER_NAME}-ng
    desiredCapacity: 2
    minSize: 1
    maxSize: 10
iam:
  withOIDC: true
EOF
```

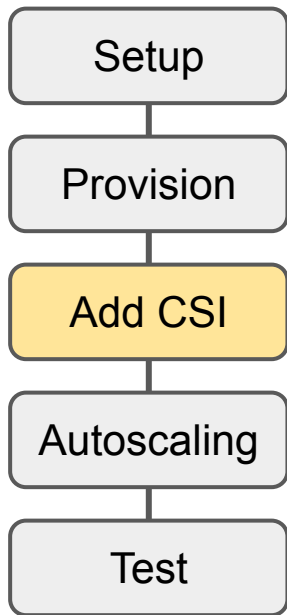
Add CSI driver to managed storage (Kubernetes 1.23+)



```
eksctl create iamserviceaccount \
  --name ebs-csi-controller-sa \
  --namespace kube-system \
  --cluster ubuntu-altinity-cloud-anywhere-demo \
  --attach-policy-arn
arn:aws:iam::aws:policy/service-role/AmazonEBSCSIDriverPolicy \
  --approve \
  --role-only \
  --role-name AmazonEKS_EBS_CSI_DriverRole

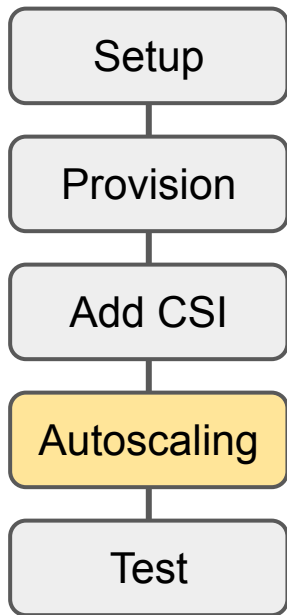
eksctl create addon --name aws-ebs-csi-driver \
  --cluster ${CLUSTER_NAME} \
  --service-account-role-arn
arn:aws:iam::${AWS_ACCOUNT_ID}:role/AmazonEKS_EBS_CSI_DriverRole \
  --force
```


Add matching storage class(es) for CSI driver



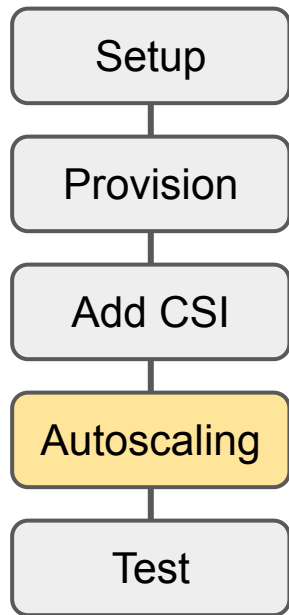
```
kubectl apply -f - <<EOF
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: gp2
  annotations:
    storageclass.kubernetes.io/is-default-class: "true"
provisioner: ebs.csi.aws.com
parameters:
  fsType: ext4
  type: gp2
reclaimPolicy: Delete
volumeBindingMode: WaitForFirstConsumer
allowVolumeExpansion: true
EOF
```

Add Karpenter to allocate/deallocate VMs automatically



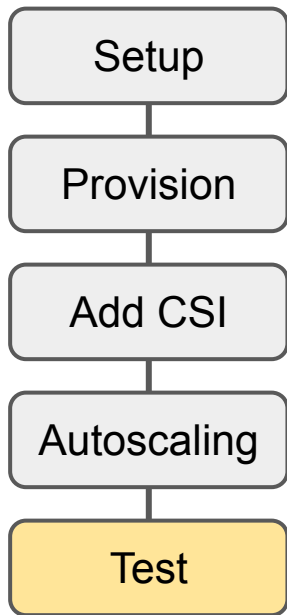
1. Install [Helm](#)
2. Install Karpenter using [getting started with eksctl](#) instructions
 - a. [Create Karpenter infra and IAM roles](#)
 - b. [Create IAM identity mapping](#) so new VMs can join cluster
 - c. [Create IAM role for KarpenterController](#) so it can launch VMs
 - d. [Install Karpenter Helm Chart](#)
 - e. [Create a Karpenter provisioner](#) to manage VMs

Here's what a Karpenter provisioner looks like



```
cat <<EOF | kubectl apply -f -
apiVersion: karpenter.sh/v1alpha5
kind: Provisioner
metadata:
  name: default
spec:
  limits:
    resources:
      cpu: 1000
  providerRef:
    name: default
  ttlSecondsAfterEmpty: 30
---
apiVersion: karpenter.k8s.aws/v1alpha1
kind: AWSNodeTemplate
metadata:
  name: default
spec:
  subnetSelector:
    karpenter.sh/discovery: ${CLUSTER_NAME}
  securityGroupSelector:
    karpenter.sh/discovery: ${CLUSTER_NAME}
EOF
```

Try it out!



Make sure your Kubernetes cluster can provision nodes and allocate storage!

Try installing a simple application like nginx and prove that

1. Karpenter can allocate a VM for it.
2. Storage allocation on block storage works.

Common problems:

- You missed IAM privileges somewhere
- CSI driver and matching storageclass missing/invalid
- Karpenter provisioner not correctly defined

Seems complicated! Isn't there a better way?

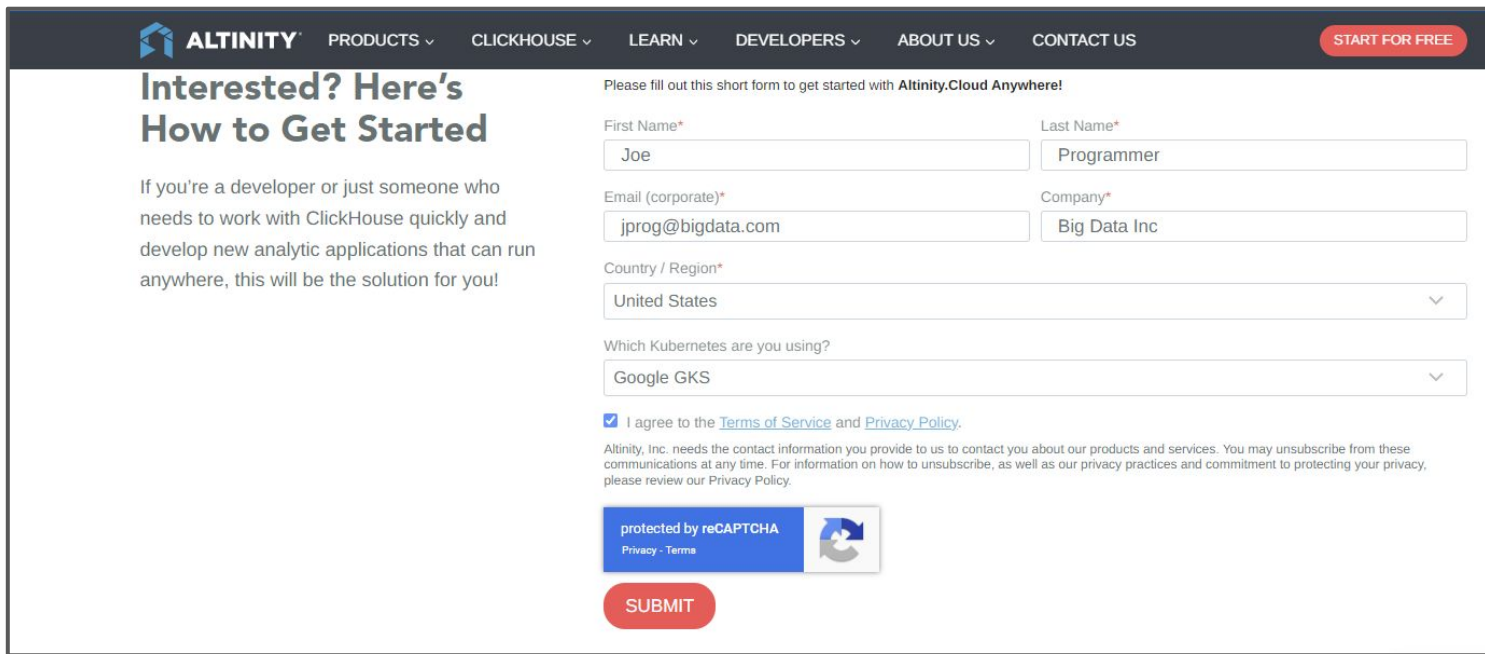
Yes!!

Starting in Q1 2023, Altinity.Cloud will offer full provisioning of EKS from soup-to-nuts.

All you need to supply is an IAM role and a VPC.

Connecting to Altinity.Cloud

Get an Altinity.Cloud account (if you don't have one)



The screenshot shows the Altinity website's sign-up page. The header is dark blue with the Altinity logo and navigation links: PRODUCTS, CLICKHOUSE, LEARN, DEVELOPERS, ABOUT US, and CONTACT US. A red 'START FOR FREE' button is on the right. The main content area has a heading 'Interested? Here's How to Get Started' and a paragraph: 'If you're a developer or just someone who needs to work with ClickHouse quickly and develop new analytic applications that can run anywhere, this will be the solution for you!'. Below this is a form titled 'Please fill out this short form to get started with Altinity.Cloud Anywhere!'. The form includes fields for First Name (Joe), Last Name (Programmer), Email (jprog@bigdata.com), Company (Big Data Inc), Country / Region (United States), and Which Kubernetes are you using? (Google GKS). There is a checkbox for 'I agree to the Terms of Service and Privacy Policy.' and a reCAPTCHA widget. A red 'SUBMIT' button is at the bottom.

ALTINITY PRODUCTS ▾ CLICKHOUSE ▾ LEARN ▾ DEVELOPERS ▾ ABOUT US ▾ CONTACT US [START FOR FREE](#)

Interested? Here's How to Get Started

If you're a developer or just someone who needs to work with ClickHouse quickly and develop new analytic applications that can run anywhere, this will be the solution for you!

Please fill out this short form to get started with **Altinity.Cloud Anywhere!**

First Name* Last Name*


Email (corporate)* Company*

Country / Region* ▾

Which Kubernetes are you using? ▾

☒ I agree to the [Terms of Service](#) and [Privacy Policy](#).

Altinity, Inc. needs the contact information you provide to us to contact you about our products and services. You may unsubscribe from these communications at any time. For information on how to unsubscribe, as well as our privacy practices and commitment to protecting your privacy, please review our [Privacy Policy](#).

protected by reCAPTCHA 

[Privacy - Terms](#)

[SUBMIT](#)

<https://altinity.com/altinity-cloud-anywhere>

Sign into Altinity.Cloud and find your environment

Connecting rhodges-anywhere-3 to Altinity.Cloud

Connection Setup

Resources Configuration

Confirmation

1. Download Altinity.Cloud connect: <https://github.com/Altinity/altinitycloud-connect>

2. Connect to Altinity.Cloud

```
altinitycloud-connect login --
token=eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJhdWQiOiIjZdGfNaW5nLmFsdGluaXR5LmNsY3VkiwiIiwiaXNjaXoOTzE1ZnUzLjZwIiwiaG9kZVZlWmFueXdoZXJ1LTMi
nRmtSN0QzQ1WGoCfEElusufQcGZnhpdxmFrP1iStI0y19UkrIyEKvoHXmfK-Ex-oqxsCB9rZDdqRkh33rr6P54v6qKiguFgljyxf920p9pWNNF0-KNzQxz0_-dt5DC1kiPBPMYB9Y8Xg-
4h6S2hz7H6HQUB8je-zfxm8AH6jPIIJMU15oVWeq1nrUKBUDFmRsdGqCHAegVekwduIxOfd_hoeLRmVOCaIBFhlyTmG3tYBi9H-
4mi3ZVkiR1k_Qln6T9v5T2jWvQ0iIvJvUPHkmEzPBK9a0FeervvHz_fCEaEcQ1SgxHm3KIKG0-BqPiInFJoN9fJgm893-nan_pLOx-
pV0ZRLry8uSulWysJSVJQH7Mr1fApEzHvfPimGp7GRB1uqRUxxJV9kVi-qP0r4dnpYm76yzanKL100MtUoNlQwZ-zoo1Dr3A0C_wabfZ-
nn99_U2zci_t_W1wF19gB9qeezI9r4LNwqLVXKno1hkRmLdafJNAKPAYaNezLIYUvzkMm1fqrDEDCq-
jj17rpX_UNiShqK52o1kIXfCKfpZd8ANDG8ZH2tDYer0dXavo8jPotdIWp12tx3vM8bcp1Vwt_b8RGTC3pNHT9MgT6jgZHGqYQPSNEUJV9T8n980J9gkn0GiHKYo
```

3. Deploy connector to your Kubernetes cluster

```
altinitycloud-connect kubernetes | kubectl apply -f -
```

PROCEED

CANCEL

<https://acm.altinity.cloud>

Install Altinity Connector and set up connection

```
# Download altinitycloud-connect.
```

```
curl -sSL
```

```
https://github.com/altinity/altinitycloud-connect/releases/download/v0.9.3/altinitycloud-connect-0.9.3-linux-amd64 -o altinitycloud-connect \
&& chmod a+x altinitycloud-connect \
&& sudo mv altinitycloud-connect /usr/local/bin/
```

```
# Login to Altinity.Cloud.
```

```
altinitycloud-connect login --token=<registration token>
```

```
# Pipe setup commands to deploy connector to your Kubernetes cluster.
```

```
altinitycloud-connect kubernetes | kubectl apply -f -
```

Fill out resource configuration

Connecting rhodges-anywhere-3 to Altinity.Cloud

Connection Setup **Resources Configuration** Confirmation

Provide Storage Classes and Node Types that are available in your environment

Cloud Provider
☐ Not Specified ☒ AWS ☐ GCP

Storage Classes

gp2

+ ADD STORAGE CLASS

Node Pools

Zone	Instance Type	Capacity	Used For		
us-west-2b	m5.large	10	<input checked="" type="checkbox"/> ClickHouse <input checked="" type="checkbox"/> Zookeeper <input checked="" type="checkbox"/> System	+	🗑️
us-west-2b	m5.large	10	<input checked="" type="checkbox"/> ClickHouse <input checked="" type="checkbox"/> Zookeeper <input checked="" type="checkbox"/> System	+	🗑️
us-west-2c	m5.large	10	<input checked="" type="checkbox"/> ClickHouse <input checked="" type="checkbox"/> Zookeeper <input checked="" type="checkbox"/> System	+	🗑️

+ ADD NODE POOL

PROCEED CANCEL

Block storage types

Node pool AZs and machine types

Services that will use the node pool

Review and Press FINISH to initiate setup

Connecting rhodges-anywhere-3 to Altinity.Cloud

Connection SetupResources ConfigurationConfirmation

Verify your resource configuration

Resources Specification JSON format

```
{  "cloudProvider": "AWS",  "storageClasses": [    {      "name": "gp2"    }  ],  "nodePools": [    {      "for": [        "CLICKHOUSE",        "ZOOKEEPER",        "SYSTEM"      ],      "instanceType": "m5.large",      "zone": "us-west-2b",      "capacity": 10    },    {      "for": [        "CLICKHOUSE"
```

FINISH

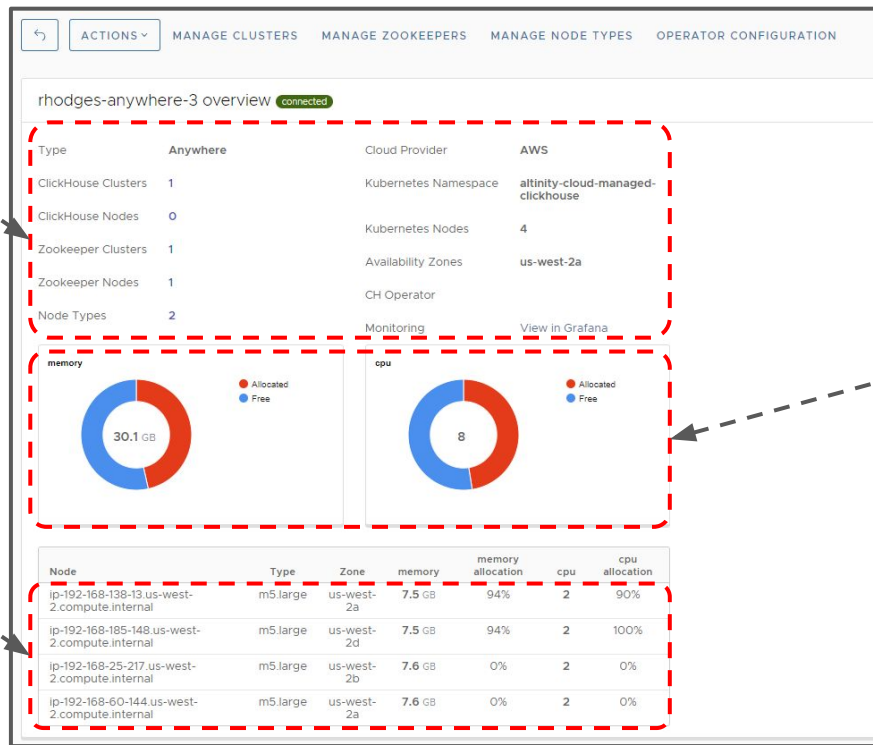
CANCEL

Altinity.Cloud Anywhere environment dashboard

Environment statistics

Current nodes

Allocated RAM and storage



Working with Clusters in Anywhere

Cluster view is the same as Altinity.Cloud

Create a new cluster

LAUNCH CLUSTER

DISCOVER

IMPORT

TOOLS

SHARE

webinar-demo

2 / 2 nodes online

Connection Details

Health	6/6 checks passed
Shards	1
Replicas	2
Storage	100 GB / node
Memory	7 GB / node
CPU	2 / node
Version	22.12.3.5
Backup	N/A

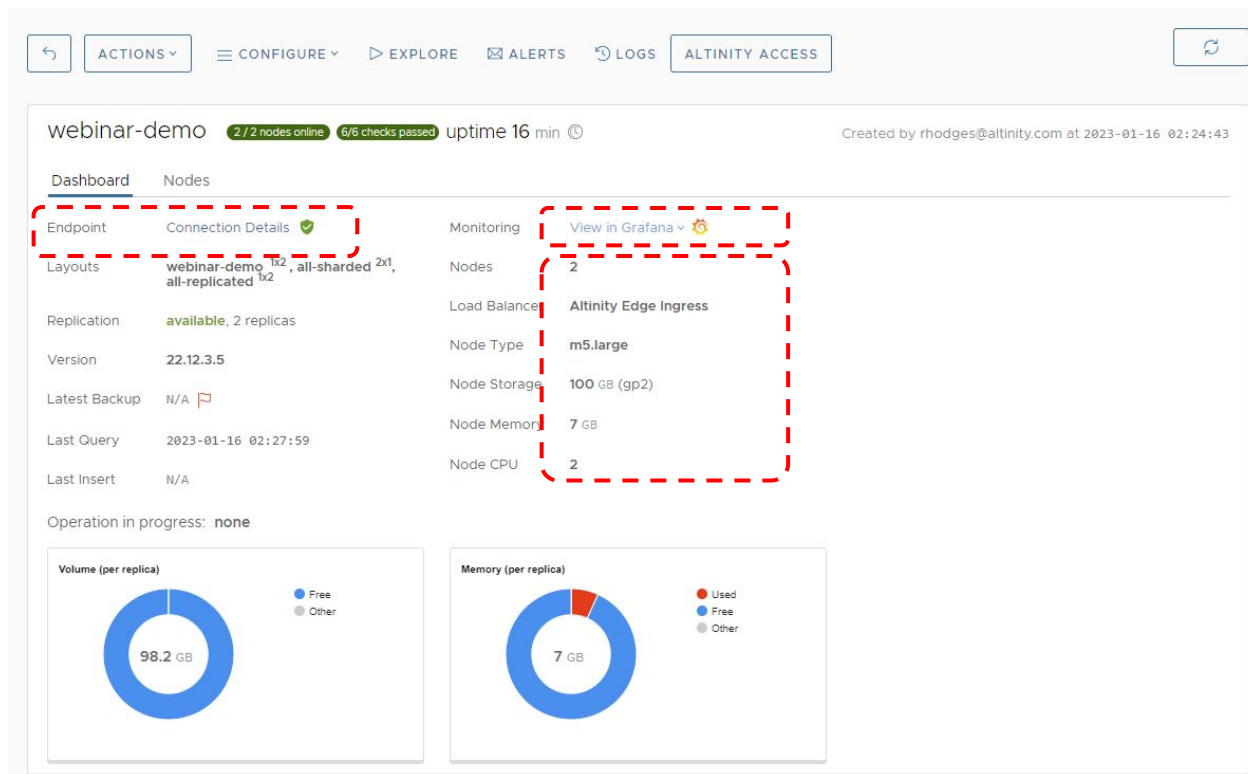
CONFIGURE

EXPLORE

ACTIONS

An active cluster

So is the cluster dashboard



Rescaling is limited by your resources

Horizontal scaling

Desired Cluster Size (edit to make changes)

Number of Shards:	1
Number of Replicas:	2

Vertical scaling

Desired Node Size (edit to make changes)

Node Type:	m5.large (CPU x2, RAM 7
Node Storage:	100
Number of Volumes:	1
Apply to new nodes only	<input type="checkbox"/>

Node storage can only be extended. The minimum increase is 20%

Rescale Cluster

Rescale options for the Cluster **webinar-demo**

Current Cluster Size

Number of Shards:	1
Number of Replicas:	2

Current Node Size

Node Type:	m5.large (CPU x2, RAM 7
Node Storage:	100
Number of Volumes:	1

Volume (per replica)

98.2 GB

Memory (per replica)

7 GB

Add new node types from the Environment Dashboard

The screenshot displays the Altinity Environment Dashboard interface. At the top, there are navigation tabs: 'MANAGE CLUSTERS', 'MANAGE ZOOKEEPERS', and 'MANAGE NODE TYPES'. Below these, the 'rhodges-anywhere-3 overview' is shown with a 'connected' status. A green '+ ADD NODE TYPE' button is visible. Below the button is a table listing existing node types:

#	Name	Scope	Code	CPU	Memory	Labels
1	m5.large	clickhouse	m5.large	2	7 GB	m5.large
2	m5.large	zookeeper	m5.large	2	7 GB	m5.large

Below the table, there are two donut charts: 'memory' showing 52.8 GB allocated (red) and 'cpu' showing 14 allocated (red). A dashed arrow points from the '+ ADD NODE TYPE' button to a 'Node Type Details' modal on the right. The modal contains the following fields:

- Name: m5.2xlarge
- Scope: Clickhouse
- Code: m5.2xlarge
- Memory, MB: 0
- CPU: 0
- Storage Class: gp2
- Advanced options: Provider Label (Enter cloud provider pool label), Extra Spec (Enter JSON-formatted specification)

The modal has 'CANCEL' and 'OK' buttons at the bottom right.

Configure backups in Environment->Edit->Backups

Environment Details

Common Options Kubernetes Options Container Options Resources **Backups** Datadog

Turn On Backups ☒

Backup Tool Image [ClickHouse Backup Image](#)
Default Image: altinity/clickhouse-backup.2.1.3

Backup Schedule
⚠ WARNING: If you change the scheduled time, then previously created backups could be deleted automatically

Period	Day of Week/Month	Time (GMT)	Backups to Keep
Daily	Sunday	05:00 AM	3

+ ADD SETTING

Compression Format tar

Backup Storage Provider ☒ AWS ☐ GCP

Access Key NARVV5SIHSTTXDXFYXZ9

Secret Key *****

Region us-west-2

Bucket * my-company-backup-bucket1

Path altinity-cloud-managed-clickhouse
Do not add trailing slash. Cluster name will be automatically appended.
If left empty, path value will be altinity-cloud-managed-clickhouse

CANCEL OK

Enable backups

Backup schedule

S3 bucket details

You can backup and restore Kubernetes data

The screenshot displays the Altinity Cloud Managed ClickHouse console interface. The 'ACTIONS' menu is open, showing options like Upgrade, Rescale, Stop, Restart, Export Configuration, Publish Configuration, Launch a Replica Cluster, Restore a Backup, Create Backup, and Destroy. Two arrows point from the 'Restore a Backup' and 'Create Backup' options to the 'Cluster Restore Wizard' and 'Cluster Backup' modal windows respectively.

Cluster Restore Wizard

1 Backup location
2 Source Cluster
3 Source Backup
4 Destination Cluster
5 Restore Summary

Backup Information: altinity-cloud-managed-clickhouse/webinar-demo

Note: It is possible to restore a complete Cluster's configuration only if a given backup contains it.

	Tag	Size	Timestamp	Configuration
<input type="radio"/>	20230117041520	436 B	2023-01-17 04:15:20	✓
<input type="radio"/>	20230117034117	436 B	2023-01-17 03:41:17	✓

2 backups

Cluster Backup

The Backup procedure for the Cluster **webinar-demo** has been scheduled. It will require some time to finish the procedure.

Note: Backup files are handled by ACM and stored separately from the cluster instances. These backup files will remain available even if you accidentally delete the cluster.

OK

Volume (per replica)

98.2 GB

Memory (per replica)

7 GB

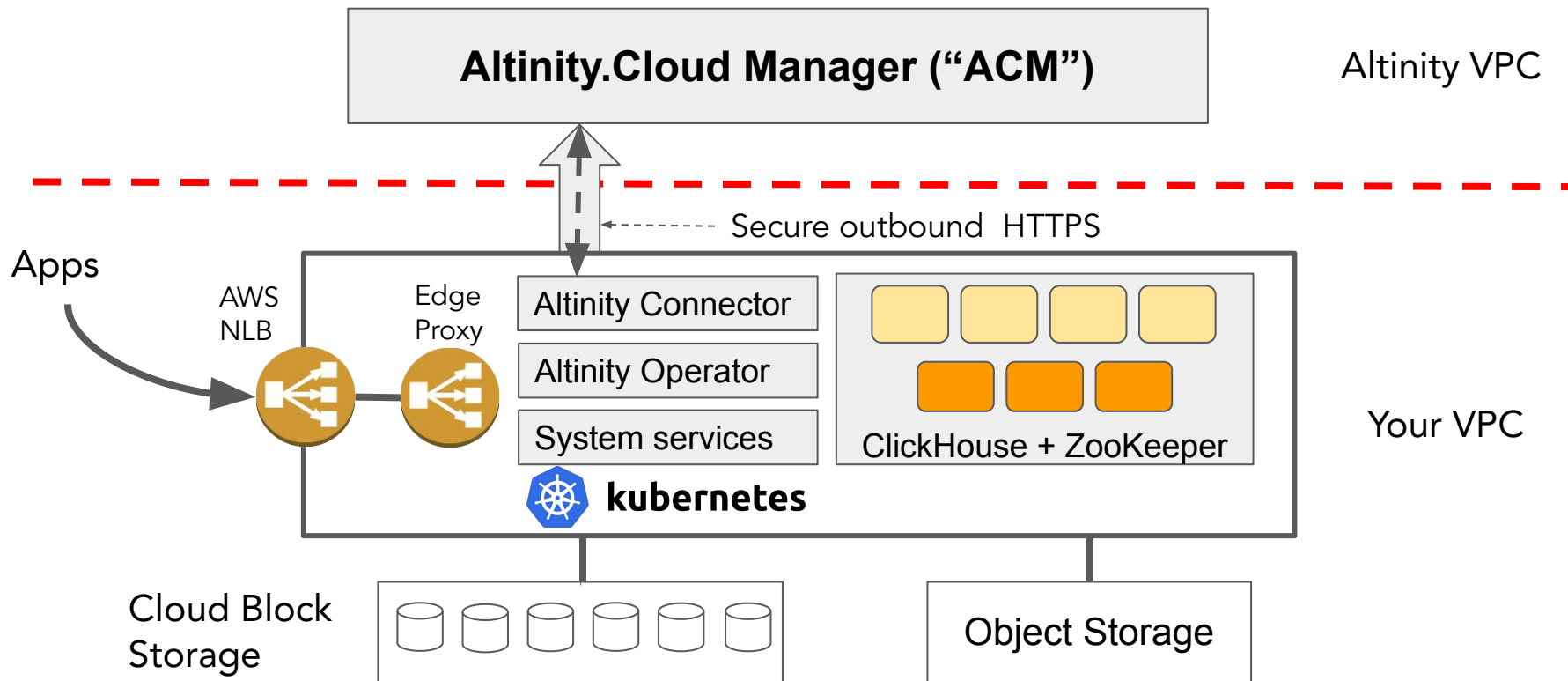
What else is there?

- Setting up private connectivity from Kubernetes to other VPCs
 - ClickHouse.Cloud supports VPC Endpoints
- Running other applications alongside managed ClickHouse
- Disaster recovery

Contact support for help on any of these!

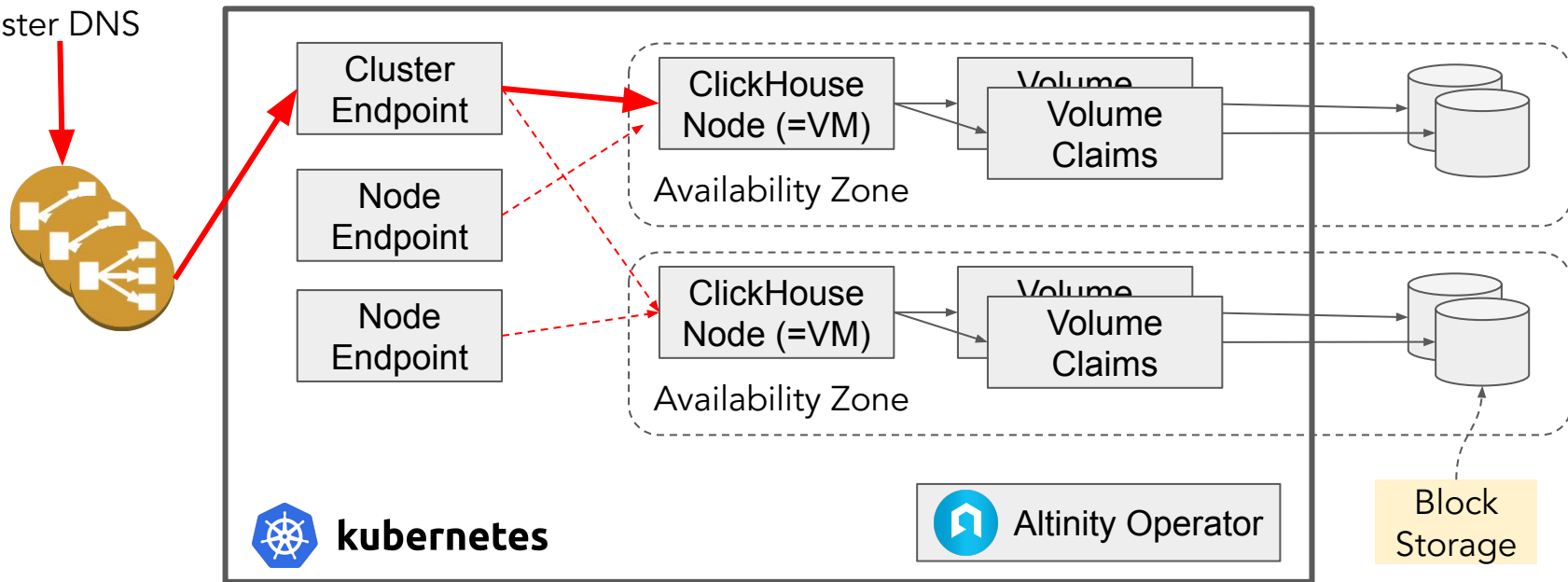
Advanced Topics

How does Altinity.Cloud Anywhere work internally?



What's going on under the covers in a ClickHouse cluster?

Connection to
Cluster DNS



Restricting access to data is a standard feature

The screenshot shows the Altinity Access configuration interface for a cluster named 'github'. The interface includes a top navigation bar with buttons for 'ACTIONS', 'CONFIGURE', 'EXPLORE', 'ALERTS', 'LOGS', and 'ALTINITY ACCESS'. A modal window is open, titled 'Altinity Access for Cluster github', showing the 'Access Level' configuration. The 'Full Access' option is selected, indicating that the Altinity User will have full access to all databases. Other options include 'No Access', 'System', and 'Read Only'. There are also checkboxes for 'Disable Cluster Configuration Management' and 'Disable Cluster Actions'. A 'CONFIRM' button is visible at the bottom of the modal. In the background, a table lists cluster components: Monitoring (with a 'View in Grafana' link), Nodes (1), Load Balancer (Altinity Edge Ingress), Node Type (m6i.8xlarge), Node Storage (1 TB (gp2)), Node Memory (122 GB), and Node CPU (32). A donut chart titled 'Memory' shows the memory usage, with a total of 122 GB. The chart is divided into 'Used' (red), 'Free' (blue), and 'Other' (grey) categories.

Altinity Access for Cluster github

Access Level

- ☐ No Access
No access provided for the Altinity User
- ☐ System
Altinity User will have **readonly** access to **system** database only
- ☐ Read Only
Altinity User will have **readonly** access to **all** databases
- ☒ Full Access
Altinity User will have full access to **all** databases

☐ Disable Cluster Configuration Management

☐ Disable Cluster Actions

CONFIRM

Component	Value
Monitoring	View in Grafana
Nodes	1
Load Balancer	Altinity Edge Ingress
Node Type	m6i.8xlarge
Node Storage	1 TB (gp2)
Node Memory	122 GB
Node CPU	32

Memory

122 GB

Legend:
● Used
● Free
● Other

How Altinity.Cloud Anywhere support works

Altinity.Cloud has enterprise support for all environments

- Schema design
- Performance optimization
- Troubleshooting
- Upgrades
- Capacity planning
- Integrations (Kafka, BI, librarise.)
- And random questions

We also answer questions about Kubernetes!

Ways to access support

- Slack - shared slack channel
- Zendesk - email to support at altinity dot com

How to get the most out of Altinity.Cloud support

- Log production issues in Zendesk, not Slack
 - Makes it easier to track progress
- Contact us in advance prior to:
 - ClickHouse version upgrades
 - Scale-out operations that may require us to raise AWS or GCP quotas
- Don't wait for problems. We can help you with migration, design and capacity planning and many other topics
 - Just ask if you have questions!
 - We also can do regular check-in calls to help you make progress

Wrap-up

- Altinity.Cloud Anywhere gives you convenience of cloud with the control of installed software
- Pricing is based on compute only. We don't charge for data
- Run ClickHouse anywhere that Kubernetes runs
- Don't want to bother with Kubernetes? We can install that, too!

Start a free two-week trial today!

<https://altinity.com/altinity-cloud-anywhere/>

Thank you! Questions?

Website: <https://altinity.com>

Email: info@altinity.com

Slack: altinitydbworkspace.slack.com

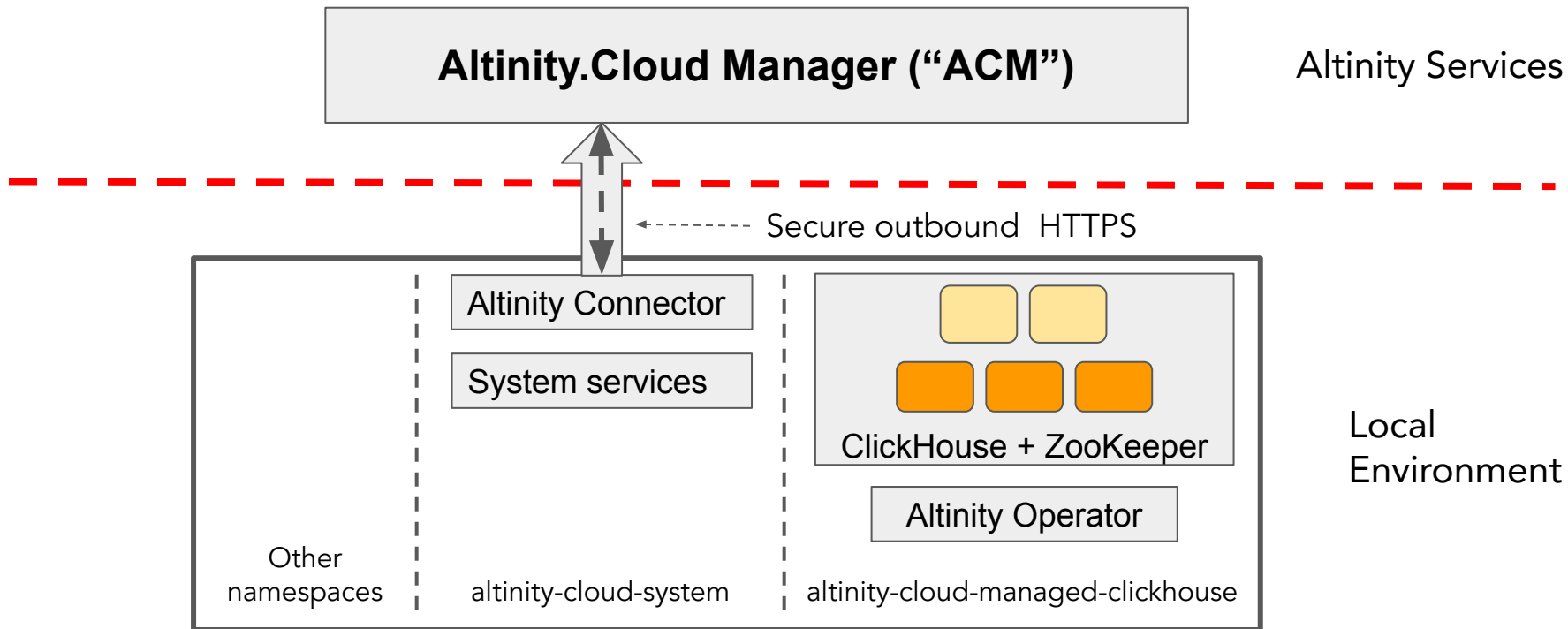
[Altinity.Cloud](#)

[Altinity Support](#)

[Altinity Stable
Builds](#)

Free 2 week trials

How does Altinity.Cloud Anywhere work internally?



Altinity.Cloud Anywhere can run in your Kubernetes

